Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Biomedical Informatics 37 (2004) 423–435

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

Comparison of character-level and part of speech features for name recognition in biomedical texts

Nigel Collier[a,*], Koichi Takeuchi[b,1]

[a] *National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan*
[b] *Okayama University, 3-1-1, Tsushima-naka, Okayama-shi, Okayama 700-8530, Japan*

Received 23 July 2004
Available online 25 September 2004

Abstract

The immense volume of data which is now available from experiments in molecular biology has led to an explosion in reported results most of which are available only in unstructured text format. For this reason there has been great interest in the task of text mining to aid in fact extraction, document screening, citation analysis, and linkage with large gene and gene-product databases. In particular there has been an intensive investigation into the named entity (NE) task as a core technology in all of these tasks which has been driven by the availability of high volume training sets such as the GENIA v3.02 corpus. Despite such large training sets accuracy for biology NE has proven to be consistently far below the high levels of performance in the news domain where *F* scores above 90 are commonly reported which can be considered near to human performance. We argue that it is crucial that more rigorous analysis of the factors that contribute to the model's performance be applied to discover where the underlying limitations are and what our future research direction should be. Our investigation in this paper reports on variations of two widely used feature types, part of speech (POS) tags and character-level orthographic features, and makes a comparison of how these variations influence performance. We base our experiments on a proven state-of-the-art model, support vector machines using a high quality subset of 100 annotated MEDLINE abstracts. Experiments reveal that the best performing features are orthographic features with *F* score of 72.6. Although the Brill tagger trained in-domain on the GENIA v3.02p POS corpus gives the best overall performance of any POS tagger, at an *F* score of 68.6, this is still significantly below the orthographic features. In combination these two features types appear to interfere with each other and degrade performance slightly to an *F* score of 72.3.

Keywords: Text mining; Support vector machines; Part of speech; Orthography

1. Introduction

The immense volume of data which is now available from experiments in molecular biology has led to an explosion in reported results. Most experimental results in the scientific literature, however, are still recorded in free-text format which requires time-consuming analysis and synthesis of the texts by human experts for understanding. Taken together with the fact that electronic versions of articles are now easily available to researchers online we are facing new challenges related to information filtering and navigation. In this context text mining has become an actively pursued goal of the bioinformatics community [1–14], i.e., the task of finding useful information by automatically mapping from unstructured text to a fully structured knowledge representation which can be stored and efficiently searched for in online databases. Several wider goals of text mining in scientific domains are now also becoming clear such as aiding in the screening of documents by

[*] Corresponding author. Fax: +81 3 3556 1916.
E-mail addresses: collier@nii.ac.jp (N. Collier), koichi@cl.it.okayama-u.ac.jp (K. Takeuchi).
URL: research.nii.ac.jp/~collier,
[1] Fax: +81 86 251 8178.

1532-0464/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2004.08.008

scientists, performing citation analysis, and in the integration of various types of specialized databases (e.g., Genbank [15], Swissprot [16], Protein Data Bank (PDB) [17,18], and Structural Classification of Proteins (SCOP) [19]) with facts contained in literature databases such as PubMed's MEDLINE [20].

A core component in each of the above tasks is the identification and classification of *named entities* in the texts. The named entity (NE) task is essentially to find the boundaries of technical terms and to classify them according to classes in a pre-determined taxonomy. The task is made more complex by several factors which are common to scientific and technical domains including the large size of the vocabulary [21], an open growing vocabulary [22], irregular naming conventions, as well as extensive cross-over in vocabulary between NE classes. The irregular naming arises in part because of the number of researchers and practitioners from different fields who are working on the same knowledge discovery area as well as the large number of entities that need to be named. Despite the best efforts of major journals to standardize the terminology, there is also a significant problem with synonymy so that often an entity has more than one name that is widely used. For example, class cross-over of terms may arise because many DNA and RNA are named after the protein with which they transcribe. This explains in part the difficulty for re-using existing term lists and vocabularies such as MeSH [23], UMLS [21] or those found in databases such as SwissProt. An additional obstacle to reuse is that the classification scheme used within an existing thesaurus or database may not be the same as the one in the users' ontology which may change from time to time as the consensus view of the structure of knowledge is refined.

This problem of NE recognition (terminology identification and classification) for biological NEs has received intense investigation [24,25,7,8,11,14,26] in the literature over the last 5 years. Due to recent improvements in the availability of large volume training sets, notably the GENIA corpus [27], for methods that use machine learning we now have a hope to achieve the same level of performance for accuracy (in the high 90's measured in *F* score) in the biology domain as in the news domain which is widely used in evaluation exercises within the natural language processing research community (e.g., the Sixth Message Understanding Conferences [28], the Seventh Conference on Natural Language Learning [29], and the Multilingual Entity Tasks (MET)). This, however, has not so far proven to be the case and it is becoming clear that more rigorous analysis of the factors that contribute to the model's performance is necessary to discover where the underlying limitations are. Our investigation in this paper reports on variations and interactions of two widely used feature types, part of speech (POS) tags and character-level orthographic features, and makes a comparison of how these variations influence performance. These features are linguistically shallow but have the advantage of being relatively cheap and accurate to assign. We base our experiments on a proven state-of-the-art model called support vector machines (SVMs) [30] (see also [31] for a good overview) and a small high quality subset of 100 MEDLINE abstracts that were used as the basis for development of the large GENIA collection but were not included in the 2000 released abstracts.

In the remainder of this paper in Section 2 we outline the data set used in our experiments and then discuss the basic advantages of SVMs together with implementation-specific issues such as the choice of feature set. In Section 3 we provide extensive results using variations of POS and orthographic features. This is followed in Section 4 by a brief discussion of the important trends that we found and their implications for future modelling.

2. Materials and methods

2.1. Data set

To show the application of SVMs to term extraction in unstructured texts related to the medical sciences we are using a collection of abstracts from PubMed's MEDLINE [20]. The MEDLINE database is an online collection of abstracts for published journal articles in biology and medicine and contains more than 14 million articles with on average 1000 articles being added daily (as of 2004). The annotated abstract collection [32] we used in our experiments is called *Bio1*[2] and comes from a sub-domain of molecular biology that we formulated by searching under the terms *human*, *blood cell*, and *transcription factor* in the PubMed database. From the retrieved abstracts 100 were randomly chosen for annotation by a human expert according to classes in a small top-level ontology. These were then annotated by a doctoral-qualified expert.

Our work has focussed on identifying names belonging to the classes shown in Table 1 and the total number of tokens in the corpus is 28,779. Example sentences from a marked up abstract are given in Fig. 1. The ontology [32] that underlies this classification scheme describes a simple top-level model which is almost flat except for the *source* class which shows locations where genetic events occur and has a number of sub-types.

For purposes of bench-marking and comparing our approach with others we have also provided results for

[2] Available from: http://research.nii.ac.jp/~collier/resources/bio1.1.xml.

Table 1
Annotatable class used in Bio1 with the number of word tokens

Class	#	Description
protein	2125	Proteins, protein groups, families, complexes, and substructures
dna	358	DNAs, DNA groups, regions, and genes
rna	30	RNAs, RNA groups, regions, and genes
source.cl	93	Cell line
source.ct	417	Cell type
source.mo	21	Mono-organism
source.mu	64	Multiorganism
source.vi	90	Virus
source.sl	77	Sublocation
source.ti	37	Tissue

Table 2
Annotatable class used in the GENIA 3.02 corpus

Class	Class
Amino acid monomer	Multi-cell organism
Atom	Mono-cell organism
Body part	Nucleotide
Carbohydrate	Other (artificial source)
Cell line	Other (organic compound)
Cell type	Other name
Cell component	Peptide
DNA family or group	Polynucleotide
DNA substructure	Protein complex
Domain or region of DNA	Protein family or group
Domain or region of protein	Protein subunit
Domain or region of RNA	RNA family or group
Individual DNA molecule	RNA substructure
Individual protein molecule	Multi-cell organism
Individual RNA molecule	Tissue
Inorganic compound	Virus
Lipid	

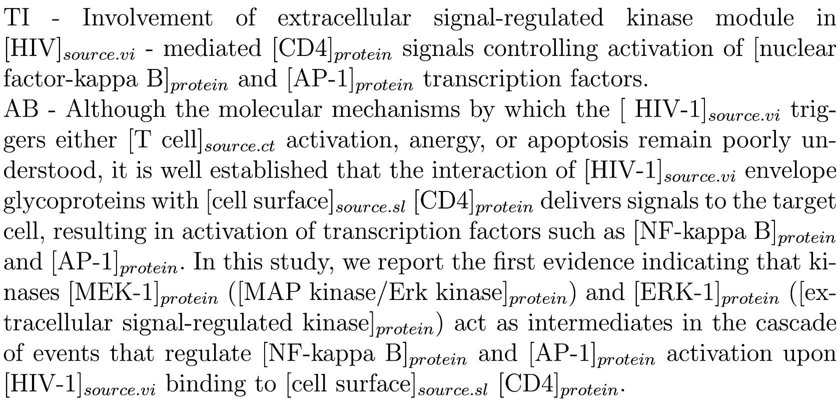
our model using combined feature sets on the GENIA version 3.02 corpus which is well documented elsewhere, e.g. [33]. Bio1 and GENIA v3.02 both come from the same source and sub-domain but do not overlap in content. The most noticeable difference between Bio1 and GENIA v3.02 is in size (2000 MEDLINE abstracts and 400,000 words, 528,113 tokens, for GENIA and 100 abstracts and 28,779 tokens for Bio1) and the number of classes (33 for GENIA v3.02 and 10 for Bio1). The 36 GENIA classes are taken from the leaf nodes of a top-level taxonomy of 48 classes based on a chemical classification. These are listed in Table 2.

The GENIA corpus is important for two major reasons: the first is that it provides the largest single source of annotated training data for the NE task in molecular biology and the second is in the breadth of classification. Although 36 classes is only a fraction of the classes contained in major taxonomies it is still the largest class set that has been attempted so far for the NE task. In this respect it is an important test of the limits of human and machine annotation capability.

2.2. Support vector machines

SVMs [34,35] have emerged as one of the leading trainable models for many classification tasks that involve discriminative learning from positive and negative examples. In their relatively short history they have been widely used in natural language processing for various tasks related to text mining including text categorization [36], noun phrase chunking [37], POS tagging [38], as well as NE recognition [39,26,40–43]. Their use has not just been confined to text mining, however, and SVMs have seen wide usage in bioinformatics [44] in tasks such as the recognition of translation initiation sites [45], protein structure predication and gene expression pattern discovery [46,47], predication of protein–protein interactions [48], and protein subcellular localization [49].

The success of SVMs is due in part to their capability to handle very large feature sets up to the order of hundreds of thousands of features [43] for capturing subtle



TI - Involvement of extracellular signal-regulated kinase module in [HIV]$_{source.vi}$ - mediated [CD4]$_{protein}$ signals controlling activation of [nuclear factor-kappa B]$_{protein}$ and [AP-1]$_{protein}$ transcription factors.
AB - Although the molecular mechanisms by which the [HIV-1]$_{source.vi}$ triggers either [T cell]$_{source.ct}$ activation, anergy, or apoptosis remain poorly understood, it is well established that the interaction of [HIV-1]$_{source.vi}$ envelope glycoproteins with [cell surface]$_{source.sl}$ [CD4]$_{protein}$ delivers signals to the target cell, resulting in activation of transcription factors such as [NF-kappa B]$_{protein}$ and [AP-1]$_{protein}$. In this study, we report the first evidence indicating that kinases [MEK-1]$_{protein}$ ([MAP kinase/Erk kinase]$_{protein}$) and [ERK-1]$_{protein}$ ([extracellular signal-regulated kinase]$_{protein}$) act as intermediates in the cascade of events that regulate [NF-kappa B]$_{protein}$ and [AP-1]$_{protein}$ activation upon [HIV-1]$_{source.vi}$ binding to [cell surface]$_{source.sl}$ [CD4]$_{protein}$.

Fig. 1. Example MEDLINE sentence marked up with square brackets to show the boundary and classes for molecular biology named-entities.

distinctions in the classifier. This capability has been empirically proven to work even when the number of patterns is relatively low.

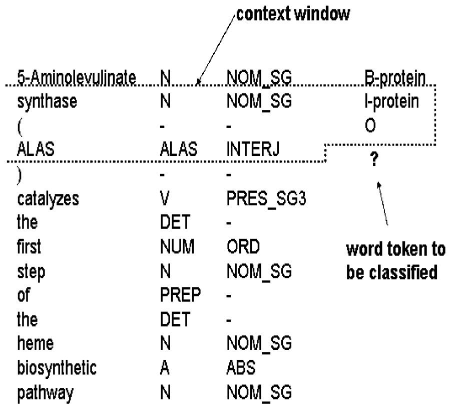
The basic approach adopted by SVMs is to construct a simple binary classification function by mapping the input patterns to a higher dimensional feature space if the patterns cannot be linearly separated in input space. The mapping function uses a dot product of the input space pattern vectors transformed according to a kernel function. The decision function implemented by the SVM is

$$f(\vec{x}) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i k(\vec{x}, \vec{x}_i) + b\right). \tag{1}$$

For the N labelled input patterns the α_i co-efficients are positive real numbers that are derived from the learning process and can be regarded as a measure of the strength with which the input pattern i is embedded in the final decision function. In fact only those patterns that contribute to defining the final decision function (called *support vectors*) will have non-zero α_i after training has completed. The final decision function will be non-linear in input space but a hyper-plane in the feature space. Moreover, the SVM learning process guarantees that the hyper-plane in maximally distant from each of the support vectors—thus minimizing a bound on testing error that comes out of Vapnik's work on statistical learning theory [34].

The kernel function we explored in our experiments was the polynomial function $k(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \cdot \vec{x}_j + 1)^d$ for $d = 2$ which was found to be the best [39] for the same task in preliminary experiments by the authors.

The cost of training SVMs using large feature sets, particularly for large training sets on higher order kernels has been found to be prohibitive (e.g. [40]). There are several working solutions to this including improvements in the modelling and internal data representations, as well as methodological improvements to detect non-contributing features before they are given to the learner. Clearly it is of benefit to the modeler to have a previous knowledge of the relative contributions of major feature types before embarking on training using complex combinations of features. The study which we report here aims to develop a more considered understanding of two of the most commonly used features available and their relative contributions to expected performance. In this respect we are less interested in absolute performance of our approach (although it does in fact compare very well with others such as [26]) than in the relative difference yielded by using different feature sets.



5-Aminolevulinate	N	NOM_SG	B-protein
synthase	N	NOM_SG	I-protein
(	-	-	O
ALAS	ALAS	INTERJ	?
)	-	-	
catalyzes	V	PRES_SG3	
the	DET	-	
first	NUM	ORD	
step	N	NOM_SG	
of	PREP	-	
the	DET	-	
heme	N	NOM_SG	
biosynthetic	A	ABS	
pathway	N	NOM_SG	

Fig. 2. Example feature window used in the classification decision.

We implemented our method using the Tiny SVM package from NAIST[3] which is an implementation of Vladimir Vapnik's SVM combined with an optimization algorithm [50]. The algorithm proceeds to classify word by word using features within the context window as shown in Fig. 2.

Before continuing it is important to note two further implementation details. The first relates to the combination of binary classifiers in order to form a multi-class model. In Tiny SVM this is accomplished by constructing $M \times (M - 1)/2$ classifiers for the M classes. The final class decision is given by majority voting. Kudoh and Matsumoto [37] in their description of Tiny SVM mention a number of advantages which motivates their choice including empirical evidence from experiments that shows a performance increase compared to other methods. The second concerns the optimization of class assignments over a sequence of word tokens in a sentence. Although dynamic programming could have been used to simulate a Viterbi-style sequence optimization algorithm this was not considered in our approach at this time. Instead we follow conventional practice by allowing the decision about the current word token to be conditioned dynamically on the previous class assignments within a fixed window.

2.3. Generalizing with features

In order for the model to be successful it must recognize regularities in the training data that relate pre-classified examples of terms with unseen terms that will be encountered in testing. For example, the learner might easily be able to infer that the string *E2F-1* was some sort of genetic product if it was told that *Saos-2* and *HIV-1* were genetic products, based on the pattern of characters such as upper case, numbers, and hyphens.

Two of the most widely used features in previous studies have been POS and character-level orthographic features. Unlike more sophisticated parsing methods which provide dependency or constituency information, both these features have the advantage of being computationally inexpensive, freely available in many forms

[3] Tiny SVM is available from http://cl.aist-nara.ac.jp/~taku-ku/software/TinySVM/.

and also powerful [7]. In this investigation we look more deeply into the relative merits of these and compare several commonly used variations.

POS define a word's lexical class (or more often class*es* in the case of ambiguous words) in a grammatical context. For example, *table* is a noun in *He found it on the table* and a verb in *He tabled the motion*. In addition to the major POS classes such as noun, verb, adjective, and adverb there are finer distinctions—the exact number varying according to the annotation scheme, e.g., 45 in the Penn Treebank (PTB) and 87 for the Brown corpus. These tags serve as the building blocks for a higher-level grammatical understanding of the text and, most important for the present study, as a clue to disambiguate the word sense and to help identify the boundaries of phrases. For an introduction to the fundamental aspects of POS tagging and pointers to further sources of general information we refer the reader to [51].

POS taggers based on supervised learning from labelled training data tend to suffer degraded performance when trained on out of domain data. It is therefore natural to expect that performance in tagging accuracy can be improved by customizing the POS tagger to a domain. The critical point though is the cost of doing this as it requires creating large amounts of hand annotated data and this must be considered against the gain in tagging accuracy. Improvements arise because of the greater overlap in vocabulary between what the tagger has seen before in training and the text currently being tagged. Also the tagger will be able to make more intelligent guesses about so-called *unknown words*, i.e., words which were never seen in the tagged training corpus, which must be tagged based on their similarities to seen words and contexts.

In our study we compared two widely used systems: the Brill tagger [52] and the Conexor FDG parser [53]. Brill is a combination of a rule-based and a stochastic-based method and is supplied with a knowledge base derived from both the Wall Street Journal and the Brown corpora using the Penn Treebank (PTB) tag set [54]. In order to explore the effects of in-domain POS tagging on NE accuracy we have taken the POS tags from the GENIA corpus (3.02p) and used these to retrain the Brill tagger. We have also derived a custom lexicon from the GENIA POS corpus for the FDG parser. In this case we converted the original POS codes, which follow a slightly modified version of the PTB tags, to their simple top-level forms such as *A* for adjectives, *N* for noun or *V* for verb. In theory the use of a special in-domain lexicon should constrain the parser in the choice it makes and thereby improve performance, e.g., by forcing it to consider *I* (as in *I kappa B*) as a common noun rather than a first person singular pronoun.

Examples can be seen in Fig. 3. Some noticeable points of difference are that the FDG POS tagger is more likely to assign an abbreviation tag to parts of protein names such as *Rel* or *B* than the FDG GENIA tagger which assigns common noun tags. The Brill WSJ tagger trained on the Wall Street Journal, like the

FDG POS

Differential_A_ABS interactions_N_NOM_PL of_PREP Rel_ABBR_NOM_SG -_- NF-kappa_N_NOM_SG B_ABBR_NOM_SG complexes_N_NOM_PL with_PREP I_PRON_PERS_NOM_SG1 kappa_N_NOM_SG B_ABBR_NOM_SG alpha_N_NOM_SG determine_V_PRES pools_N_NOM_PL of_PREP constitutive_A_ABS and_CC inducible_A_ABS NF-kappa_N_NOM_SG B_ABBR_NOM_SG activity_N_NOM_SG ._.

FDG GENIA

Differential_A_ABS interactions_N_NOM_PL of_PREP Rel_N_NOM_SG -_- NF-kappa_N_NOM_SG B_N_NOM_SG complexes_N_NOM_PL with_PREP I_N_NOM_SG kappa_N_NOM_SG B_N_NOM_SG alpha_N_NOM_SG determine_V_PRES pools_N_NOM_PL of_PREP constitutive_A_ABS and_CC inducible_A_ABS NF-kappa_N_NOM_SG B_N_NOM_SG activity_N_NOM_SG ._.

Brill WSJ POS

Differential_JJ interactions_NNS of_IN Rel_NNP -_- NF-kappa_NNP B_NNP complexes_NNS with_IN I_PRP kappa_NN B_NNP alpha_JJ determine_VB pools_NNS of_IN constitutive_JJ and_CC inducible_JJ NF-kappa_JJ B_NNP activity_NNP ._.

Brill GENIA POS

Differential_JJ interactions_NNS of_IN Rel_NN -_- NF-kappa_NN B_NN complexes_NNS with_IN I_NN kappa_NN B_NN alpha_NN determine_VB pools_NNS of_IN constitutive_JJ and_CC inducible_NN NF-kappa_NN B_NNP activity_NN ._.

Fig. 3. Illustration of the differences between POS assignments. Note that NE annotations are not shown.

FDG POS tagger, assigns the token *I* a tag for first person singular pronoun unlike the FDG GENIA tagger and the retrained in-domain Brill GENIA tagger. In contrast to the other taggers the Brill WSJ tagger mistakenly considers the second mention of NF-kappa to be an adjective rather than a singular noun. This is corrected when Brill is retrained on GENIA. In general the annotation scheme for FDG is more detailed than that for Brill with 16 main POS tags and a large number of minor ones for use in combination with the main ones making a total of 140 variations versus 46 tags for Brill. Returning to the example we notice that in the PTB annotation scheme for Brill there is no provision for considering abbreviations.

Previous NE studies that have used POS in news and molecular biology have reported mixed findings and have generally concluded that it offers little benefit or degrades performance. For example, Bikel et al. [55] found in early experiments on the news domain using a backoff hidden Markov model (HMM) that POS hid the signal from more informative orthographic features. Nobata et al. [56] found performance on a biology NE task to be degraded using an out of domain trained POS tagger with a decision tree (C4.5) model. In a further study using decision trees (C4.5) on news texts with the Brill tagger trained in-domain, Baluja et al. [57] show that although POS features individually perform better than orthographic features, in combination dictionary and word-level features outperform POS in almost every context window size.

Two recent studies [41,58], however, have shown that POS tags derived from GENIA v3.0 have led in some cases to very large improvements in performance of as much as 22.8 points of *F* score in Zhou et al. The impact of POS in these studies seems to have been to improve boundary identification which is notably difficult for biological entities where the names are often highly descriptive leading to uncertainties about where the left boundary should be placed. This implies a complex interaction between the boundary identification and classification tasks as the ambiguity regarding a term's boundary will in some cases be dependent on the class to which it has been assigned.

There is one further thought to add to these studies. This is that there is a clear difference between NEs in the news domain where the entities of interest are mostly proper nouns and character-based features have been shown to outperform POS, e.g., in the identification of words with initial capitals. This hints that POS should make a greater contribution in biology due to its contribution in the detection of noun phrase constituency boundaries. We can draw an analogy here with work done in multilingual NE in which POS has been found to make a significant contribution to German news NE [59] and there are known problems with arbitrarily long nominalizations.

Returning to feature types, orthographical information has been widely used in most NE systems both in molecular biology and in the news domain. We have derived a small set of orthographic features from earlier studies [60,7,58] and shown in Table 4 ranked according to their probabilities of predicting a class (values at the top and bottom of the table are more informative). We notice that while none of the feature values uniquely predict a particular NE class they are still in almost all cases except *TwoCaps* and *InitCap* very strong indicators, e.g., a word tagged as a string of *CapsAndDigits* has a 513 chance in 577 (88%) of being one of five possible NE classes. The clue becomes even stronger for *GreekLetter* where a token has a 145 chance in 151 (96%) of belonging to just three possible NE classes. As we would hope also, one of the feature values *LowerCase* corresponds strongly to none NEs, with a chance of 15,942 in 17,991 (89%) that the token is not an NE.

Orthographical information allows strings to be compared based on their spelling characteristics as shown in the example at the beginning of this section, this should be contrasted to phonological information which compares strings based on their *sound*. The orthographic features are contrasted with phonological features from the Soundex algorithm. Soundex is the oldest of the phonetic string matching algorithms, originally developed by Odell and Russell and used traditionally in applications involving name matching [61]. The algorithm uses codes based on the sound of each letter to map a surface string into a canonical form of up to four characters including the first letter of the string. The algorithm is in Fig. 4 using the phonetic lookup Table 3 and the implementation we used comes from the version in the Perl programming language library. Due to their sim-

1 Begin with the first letter of the string and add a 3-digit code that represents the first three remaining consonants
2 Eliminate any adjacent repetitions of codes.
3 Eliminate all occurrences of code 0 (i.e. vowels)
4 Add trailing zeroes to the result so that there are at least four characters in the resulting string
4 Return the first four characters of the resulting string.

Fig. 4. The Soundex algorithm.

Table 3
Soundex lookup table of phonetic codes

Code	0	1	2	3	4	5	6
Letters	aeiouyhw	bpfv	cgjkqsxz	dt	l	mn	r

Table 4
Orthographic feature values with examples, showing (a) the number of content classes in which the value was used—ignoring differences of the beginning and inside NE, (b) the number of NE tokens that were tagged with the value, (c) the number of non-NE tokens that were tagged with the value, and (d) probability of a value predicting a content class taken as b/(b + c)

Feature	Example	a	b	c	d
GreekLetter	kappa	3	145	6	0.96
CapsDigitHyphen	Oct-1	6	560	24	0.96
CapsAndDigits	STAT1	5	514	63	0.91
SingleCap	B	5	442	49	0.90
LettersAndDigits	p105	2	186	21	0.90
LowCaps	pre-BI	5	149	30	0.83
OneDigit	2	4	62	24	0.72
TwoCaps	EBV	8	975	505	0.66
InitCap	Sox	7	302	843	0.26
HyphenDigit	95-	2	6	36	0.14
LowerCase	kinases	10	2049	15,942	0.11
HyphenBackslash	-	5	65	530	0.11
Punctuation	(	4	118	2404	0.05
DigitSequence	98401159	1	1	135	0.01
TwoDigit	37	0	0	37	0
FourDigit	1997	0	0	4	0
NucleotideSequence		0	0	0	0

plicity both types of orthographic features are computationally inexpensive to calculate making their use advantageous for many applications.

The intuition for the use of Soundex in the NE task is the same for its use in many other applications, i.e., that we want to capture the fact that phonetically similar but orthographically variant name forms should indicate similar objects. For example, variants on the protein name *JAK* such as *JAKs*, *JAK1*, and *JAK3* all receive the same code J200 as do variants of *STAT* and LMP (see Fig. 4).

Additionally we have looked at the effects of determinism, i.e., choosing just one orthographic feature value, and have modified Collier et al.'s approach [7] to allow for a set of non-deterministic (conjoined) orthographic tags. For example, whereas the token *p105-p50* would receive only a single orthographic tag of *LettersAndDigits* in the deterministic lookup table it would receive orthographic tags *LettersAndDigits* as well as *Hyphen* in the non-deterministic version. In the approach we tried the non-deterministic features are encoded in a single binary pattern rather than as individual features.

3. Results and discussion

3.1. Assessment

Results are given as F scores [62] using the CoNLL evaluation script and are defined as $F = (2PR)/(P + R)$. where P denotes Precision and R Recall. P is the ratio of the number of correctly found NE chunks to the number of found NE chunks, and R is the ratio of the number of correctly found NE chunks to the number of true NE chunks. All results are calculated using 10-fold cross-validation using a variety of context windows given generally as $-n + m$ where n and m, respectively, show the size of the context window to the left and right of the token under consideration. For example, -10 provides features for the previous and current word, and $-1 + 1$, provides features for the previous word, current word and next word, and so on. This is to show the effects of the context window on the classifier's performance. The baseline model (shown as *base*) includes surface word, lemma, and the previous two class assignments. It should be noted that the two previous class assignments are used throughout all experiments and are not changed according to the context window size.

3.2. Comparison of character-level features

Precision, recall, and F scores for large-scale experiments are shown in Table 5 for the base model in combination with deterministic and non-deterministic orthographic features as well as Soundex features. If we focus on the results at the largest data size used in the experiments we see that the best performing model is *BaseNDO* (Base plus non-deterministic orthographic

Table 5
Precision, recall, and F scores on Bio1 showing the effects of training set size, character-level feature sets, and context window sizes

Feature set and window size	Percentage of data used in experiment				
	20	40	60	80	100
Base −10	66.6,38.1 48.5	69.5,41.7 52.1	68.5,41.1 51.4	68.5,42.1 52.1	70.9,47.2 56.7
Base −1 + 1	68.7,36.1 47.4	73.5,42.5 53.8	74.5,44.8 56.0	76.5,46.9 58.1	77.8,52.0 62.3
Base −2 + 2	72.6,27.6 40.0	76.3,37.1 50.0	75.2,40.7 52.8	76.1,44.7 56.3	78.5,49.8 60.7
Base −3 + 3	74.7,21.5 33.3	74.9,30.3 43.1	72.0,34.3 46.5	74.0,38.6 50.8	76.9,44.6 56.4
BaseDO −10	62.5,56.1 59.1	67.2,62.6 64.8	65.5,62.4 63.9	64.6,62.9 63.7	66.2,64.6 65.4
BaseDO −1 + 1	66.2,56.4 60.9	71.2,63.1 66.9	71.1,65.9 68.4	71.6,67.2 69.4	74.2,70.6 72.3
BaseDO −2 + 2	65.7,51.8 57.9	72.0,62.7 67.1	70.5,63.2 66.6	72.7,66.0 69.2	74.9,69.1 71.9
BaseDO −3 + 3	64.4,48.7 55.5	67.8,57.9 62.5	68.8,60.6 64.4	71.7,63.6 67.4	74.2,67.0 70.4
BaseNDO −10	63.4,54.8 58.8	66.6,61.8 64.1	64.7,61.5 63.1	64.6,62.6 63.6	67.2,65.4 66.3
BaseNDO −1 + 1	67.3,55.6 60.9	71.7,63.9 67.6	71.3,65.7 68.4	72.0,67.4 69.7	74.7,70.6 72.6
BaseNDO −2 + 2	67.0,52.0 58.6	71.6,62.0 66.5	70.9,63.1 66.7	73.2,66.2 69.6	75.3,69.0 72.0
BaseNDO −3 + 3	64.9,47.4 54.8	67.4,57.5 62.0	68.8,60.1 64.2	71.0,62.4 66.4	74.5,66.8 70.4
BaseSoundex −10	65.4,38.1 48.2	68.9,44.3 53.9	68.5,45.4 54.6	68.8,46.5 55.5	72.3,52.6 60.9
BaseSoundex −1 + 1	70.5,38.3 49.6	75.0,46.0 57.0	75.3,48.3 58.9	77.6,51.5 61.9	79.5,57.0 66.4
BaseSoundex −2 + 2	72.0,28.1 40.4	78.3,38.4 51.5	75.1,42.6 54.4	77.3,47.1 58.5	79.9,52.7 63.5
BaseSoundex −3 + 3	74.3,21.5 33.3	75.4,31.1 44.1	71.5,35.3 47.3	74.6,40.1 52.2	77.6,46.8 58.4

Base: surface word, lemma, and previous two SVM class assignments; DO: deterministic orthographic features; NDO: non-deterministic orthographic features; and Soundex: Soundex features.

features) with a −1 + 1 window. This slightly outperforms the deterministic model by 0.3 points of F score with a −1 + 1 window. Since the size of the data set makes accurate comparisons at this level difficult we make only a tentative conclusion from this.

With regard to a comparison with Soundex we find that while the use of phonetic features has improved performance consistently above the baseline, the phonetic features do not offer the same contribution to performance as the orthographic features. In error analysis one reason we found for this was the wide range of surface forms that could be included in some Soundex codes. For example, the algorithm discovers spurious relationships such as the code I536 which includes *interleukin-2*, *interactions*, *interact*, and *intermediates*. While Soundex does help discover some useful orthographic variations such as those described earlier for protein abbreviations, it seems also to introduce noise. This supports an earlier study [63] that report the poor applicability of phonetic matching methods such as Soundex and a variant called Phonix to string matching in an information retrieval task setting.

A further point that we note from the precision and recall figures for the *Base* model is that as the context window increases both precision and recall improve moving from −10 to the −1 + 1 context windows. Once this optimal model is reached, however, recall falls rapidly indicating over-fitting of the model due to the highly specific contexts seen in training. This is particularly prominent when we observe that precision keeps on rising with more specific contexts. This underlying trend is present in all the other results for *BaseDO*, *BaseDNO*, and *BaseSoundex* although here too we observe that precision also degrades in these models while recall degrades less quickly.

3.3. Comparison of part of speech features

Results for comparing the contribution made by FDG, GENIA, and Brill POS are shown in Table 6.

Table 6
Precision, recall, and F scores on Bio1 showing the effects of training set size, POS feature sets, and context window sizes

Feature set and window size	Percentage of data used in experiment				
	20	40	60	80	100
Base −10	66.6,38.1 48.5	69.5,41.7 52.1	68.5,41.1 51.4	68.5,42.1 52.1	70.9,47.2 56.7
Base −1 + 1	68.7,36.1 47.4	73.5,42.5 53.8	74.5,44.8 56.0	76.5,46.9 58.1	77.8,52.0 62.3
Base −2 + 2	72.6,27.6 40.0	76.3,37.1 50.0	75.2,40.7 52.8	76.1,44.7 56.3	78.5,49.8 60.7
Base −3 + 3	74.7,21.5 33.3	74.9,30.3 43.1	72.0,34.3 46.5	74.0,38.6 50.8	76.9,44.6 56.4
BaseFDG −10	61.6,38.5 47.4	66.4,43.1 52.2	65.4,42.7 51.7	66.2,43.8 52.7	68.8,48,7 57.1
BaseFDG −1 + 1	61.6,40.8 49.1	68.9,50.2 58.1	70.1,55.0 61.7	72.7,57.6 64.3	74.5,61.7 67.5
BaseFDG −2 + 2	64.5,39.3 48.8	67.1,47.8 55.8	68.7,54.3 60.7	72.4,57.1 63.8	73.5,60.1 66.2
BaseFDG −3 + 3	61.5,33.5 43.4	62.4,43.1 51.0	65.7,49.8 56.6	69.0,62.6 59.7	72.6,57.8 64.4
BaseFDG(GENIA) −10	64.6,40.5 49.8	67.1,42.1 51.7	67.0,42.5 52.0	67.5,44.0 53.3	70.1,49.1 57.8
BaseFDG(GENIA) −1 + 1	65.0,40.6 50.0	67.4,49.7 57.2	68.3,55.1 61.0	70.8,57.9 63.7	74.1,62.0 67.5
BaseFDG(GENIA) −2 + 2	63.2,36.7 46.4	66.2,47.9 55.5	67.9,54.1 60.2	71.8,57.2 63.7	73.4,60.4 66.3
BaseFDG(GENIA) −3 + 3	63.5,33.7 44.0	63.6,44.0 52.1	66.4,51.1 57.8	68.8,53.0 59.9	72.1,57.9 64.2
BaseBrill(WSJ) −10	64.1,38.2 47.8	67.5,41.6 51.5	67.5,42.3 52.0	68.0,46.6 53.2	70.8,48.9 57.9
BaseBrill(WSJ) −1 + 1	70.0,39.5 50.5	69.1,47.7 56.4	69.7,52.3 59.8	71.0,55.7 62.5	74.4,60.9 67.0
BaseBrill(WSJ) −2 + 2	66.2,34.2 45.1	67.7,45.5 54.4	69.6,53.0 60.2	70.9,55.4 62.2	73.8,59.8 66.1
BaseBrill(WSJ) −3 + 3	63.8,29.6 40.5	63.7,40.7 49.7	66.7,48.1 55.9	69.1,51.1 58.7	72.0,55.3 62.6
BaseBrill(GENIA) −10	63.9,40.3 49.4	67.7,42.3 52.5	66.0,43.0 52.1	68.1,44.5 53.8	70.8,49.4 58.2
BaseBrill(GENIA) −1 + 1	67.0,42.6 52.1	69.9,50.4 58.6	72.0,56.2 63.1	72.4,57.8 64.3	75.2,63.0 68.6
BaseBrill(GENIA) −2 + 2	65.1,36.6 56.9	68.4,47.1 55.8	70.4,54.6 61.5	73.4,57.0 64.2	74.4,61.6 67.4
BaseBrill(GENIA) −3 + 3	65.6,32.7 43.6	65.4,42.8 51.8	67.8,50.6 57.9	70.7,52.3 60.1	72.8,57.5 64.3

Base: surface word, lemma, and previous two SVM class assignments; DO: deterministic orthographic features; NDO: non-deterministic orthographic features; Soundex: Soundex features; FDG POS: FDG POS features; FDG(GENIA): FDG POS supplemented by a custom lexicon from GENIA v3.02p; Brill(WSJ): Brill POS from PTB; and Brill(GENIA): Brill POS from GENIA v3.02p.

In almost every case the model with context window $-1+1$ performs better than any other. The trend for increasing sizes of training data remains consistently that the Brill GENIA tags provide the best performance followed by FDG and FDG with GENIA. The overall difference between FDG and FDG with GENIA is not very large, at approximately a 0.56 per cent drop in F score at 100 per cent data. The relative differences between both FDG models and Brill though is more obvious, e.g., at 100 per cent of data Brill GENIA performs 2.2 per cent better than FDG GENIA and 2.4 per cent better than Brill WSJ.

3.4. Combining part of speech with complex feature sets

We observed earlier that it is not simply enough to measure the effects of individual features used in isolation as the features interact in complex ways in the SVM learning process. For this reason we have tested performance using a complex feature set plus each of the POS schemes.

Results are shown in Table 7 and indicate agreement with earlier studies showing that POS and orthographic features do not mix well. F scores for *BaseNDO Brill(GENIA)* at 72.3 on a $-1+1$ context window are

Table 7
Precision, recall, and *F* scores on Bio1 showing the effects of training set size, POS feature sets, and context window sizes

Feature set and window size	Percentage of data used in experiment				
	20	40	60	80	100
BaseNDOBrill(GENIA) −10	62.7,54.6	66.6,61.3	66.0,62.2	64.9,62.2	66.7,64.4
	58.4	63.9	64.1	63.5	65.5
BaseNDOBrill(GENIA) −1 + 1	64.6,54.5	70.3,63.6	71.4,66.0	71.8,66.9	74.1,70.5
	59.1	66.8	68.6	69.3	72.3
BaseNDOBrill(GENIA) −2 + 2	65.6,51.8	71.6,61.4	71.0,64.4	72.3,65.7	75.7,69.4
	57.9	66.1	67.5	68.8	72.4
BaseNDOBrill(GENIA) −3 + 3	65.3,48.8	67.0,57.5	68.5,60.7	71.1,63.2	74.3,67.0
	55.9	61.9	64.3	67.0	70.5
BaseSoundexBrill(GENIA) −10	63.1,40.1	68.6,48.7	68.1,49.7	67.4,49.8	70.2,55.4
	49.0	57.0	57.5	57.3	61.9
BaseSoundexBrill(GENIA) −1 + 1	68.7,43.1	72.3,52.2	72.7,56.6	73.7,58.7	76.0,63.3
	52.9	60.6	63.7	65.4	69.1
BaseSoundexBrill(GENIA) −2 + 2	67.0,36.0	71.1,48.4	71.8,54.4	74.6,57.4	75.8,61.1
	46.8	57.6	61.9	64.9	67.7
BaseSoundexBrill(GENIA) −3 + 3	69.5,32.0	67.2,42.8	69.5,50.1	72.7,52.4	75.0,58.3
	43.8	52.3	58.2	60.9	65.6
BaseNDOSoundex −10	63.3,55.0	67.5,63.3	64.8,61.8	65.2,62.9	67.6,65.6
	58.8	65.3	63.3	64.0	66.6
BaseNDOSoundex −1 + 1	68.7,57.1	71.7,64.1	71.1,65.8	72.0,67.2	75.1,70.9
	62.3	62.7	68.4	69.5	73.0
BaseNDOSoundex −2 + 2	69.3,51.0	72.3,62.3	71.1,63.7	73.1,66.3	75.5,68.8
	58.8	66.9	67.2	69.5	72.0
BaseNDOSoundex −3 + 3	68.1,45.9	67.5,56.7	68.5,59.8	71.1,62.8	74.6,66.8
	54.8	61.6	63.8	66.7	70.5

Base: surface word, lemma, and previous two SVM class assignments; DO: deterministic orthographic features; NDO: non-deterministic orthographic features; Soundex: Soundex features; FDG POS: FDG POS features; FDG(GENIA): FDG POS supplemented by a custom lexicon from GENIA v3.02p; Brill(WSJ): Brill POS from PTB; and Brill(GENIA): Brill POS from GENIA v3.02p.

below the 72.6 achieved by the *BaseNDO* model alone. The best result for *BaseNDOBrill(GENIA)* is seen at a −2 + 2 context window and is still below the best reported *BaseNDO* result. Surprisingly though the best result of mixing features comes from *BaseNDOSoundex* at a −1 + 1 context window with 73.0 *F* score.

The trend which we observed is consistent with the observations made in [55,64] which is that POS seems to contribute significantly less than orthographic features. From Tables 5 and 6 it seems clear the orthographic features have a strong advantage in gaining recall with little loss in precision, thus contributing to comparatively high *F* scores. For all of the POS taggers the final *F* score at 100 per cent of data using the best context window is below all but one of the orthographic feature models in Table 5 using a comparable training data set and context window.

3.5. Benchmark GENIA test

In order to make our results comparable with others such as [26,58] we have provided benchmark results for the GENIA version 3.02 collection in Table 8. While it is not possible to directly compare our results with those of Kazama et al. who used an earlier version of the GENIA corpus and achieving about 54.4 *F* score with an SVM we believe that our results (65.4 using a rich feature set) reflect well against the state-of-the-art in this area. The results reported by Zhoe et al. of 66.6 *F* score on GENIA v3.0 using a variant of the hidden Markov model enhanced for rich feature sets are the closest comparable to ours although they used a simplified subset of the 33 classes by ignoring subclasses of protein, DNA, and RNA. This is reasonable due to possible human annotation errors which they mention in their article, but we wanted to test our system against human performance on all classes, even where there is noise.

3.6. Discussion

The results presented so far have shown empirically some interesting trends in the use of orthographic and part of speech features. In order to get a deeper insight into what is happening we performed error analysis on the output of *BaseNDO −1 + 1* and *BaseBrill(GENIA) −1 + 1*. Several points emerged:

- In many cases where there is a noun phrase involving three or more word tokens, there is a complex relationship between syntax and semantics caused by the ontological view. This means that the right- or left-hand boundary is not always easy to define

Table 8
Benchmark *F* scores on GENIA v3.02 showing the effects of training set size, complex features sets including POS features, and context window sizes

Feature set and window size	Percentage of data used in experiment				
	20	40	60	80	100
Base −2 + 2	51.4	54.6	56.6	57.5	58.0
BaseNDO −2 + 2	54.0	56.5	58.2	59.1	59.4
BaseSoundex −2 + 2	52.2	55.3	57.2	58.1	58.7
BaseFDG −2 + 2	53.4	56.1	57.8	58.6	59.0
BaseFDG(GENIA) −2 + 2	52.8	55.4	54.4	58.2	58.7
BaseBrill(GENIA) −2 + 2	53.8	56.6	58.0	59.2	59.6
BaseBrill(WSJ) −2 + 2	55.1	57.2	59.0	59.7	60.1
BaseFDGOthers −1 + 1	62.8	64.6	65.2	65.7	65.4

Base: surface word, lemma, and previous two SVM class assignments; DO: deterministic orthographic features; NDO: non-deterministic orthographic features; Soundex: Soundex features; and FDG POS: Others: head noun and dependency features from the FDG parser, deterministic orthographic feature and Soundex feature.

despite syntactic evidence from the POS tagger. For example, in "[early B-cell]$_{source.ct}$ development" even though the Brill tagger identifies the NP boundary correctly, "development" should not be annotated as the full NP is not an instance of any of the classes. A similar case is found with ["beta-globin gene]$_{DNA}$ activation."

- "[EKLF]$_{protein}$ null [mice]$_{source.mu}$," shows a different problem whereby the annotator has chosen to ignore "null." We find this again with " [Pax-5]$_{protein}$ mutant [bone marrow]$_{source.ti}$."
- A further problem is where the noun phrase contains an embedding of words belonging to different classes and the annotation scheme requires us to find the outer most bracket. For example, "[transferrin receptor mRNAs]$_{RNA}$" contains a potential "[transferrin receptor]$_{protein}$." Also "[T cell enhancer]$_{protein}$ contains a potential "[T cell]$_{source.ct}$." Neither of these inner names are required to be annotated in the simple annotation scheme but are found and wrongly assigned by both models.
- In most cases word feature information (i.e., the surface form of the word itself) is the most informative but this sometimes biases the classifier into a wrong decision where the expression being considered is a rare case. For example in "[NF-kappa B motif]$_{DNA}$," the word "motif" appears tagged as a DNA whereas in most of the rest of the corpus it is not tagged as a content class. Both models failed to recognize this correctly and close the right-hand boundary at "B."
- On the whole the orthographic models were more aggressive at assigning content classes where the Brill (GENIA) model chose to not assign them.
- Long phrases remain a source of low accuracy for both models. For example, neither assigned any label to "[Schizosaccharomyces pombe Mc mating type gene]$_{DNA}$."
- Phrases which contain embedded abbreviations are a problem and require special treatment to resolve local syntactic ambiguities. For example, "[human immunodeficiency virus 1]$_{source.vi}$ ([HIV-1]$_{source.vi}$) [long terminal repeat]$_{DNA}$ ([LTR]$_{DNA}$)." Both models made surprisingly good attempts at this but the Brill (GENIA) model mis-tagged "[human]$_{source.mu}$."

4. Conclusion

Large-scale evaluation of the influence of various orthographic and POS features has revealed some insightful trends which should make the work of language modelling easier for the NE task in the molecular biology domain. Simple orthographic features have consistently been proven to be a valuable contribution to the classification performance of most models either used in combination or separately. This is in contrast to simple phonetic algorithms like Soundex which seem to introduce an element of noise into the model.

POS appears to be less useful than orthography due to the complex relationship between name boundaries, local syntactic ambiguities, and class semantics and has shown to detract from accuracy in combination with orthography. The question remains though about why some authors have observed improved performance. In the case of Zhou et al. for which we have the clearest analysis there may be several possible reasons: the first is that there are strong influences from the POS tagging algorithm. In their experiments they tried nine different POS tagging models, all of which were trained on GENIA v3.0p. We have increased this coverage of models to Brill and FDG but we did not observe the large increases in performance that they noted in the NE tagger. The second possibility which we must conclude is the reason is that their NE tagger could incorporate the evidence from POS and combine it in a highly sophisticated way with that from the other feature types, including orthographic information. Zhoe et al. approach this by using a k-NN algorithm to resolve the problem of a fragmented probability space when their HMM is faced with large feature sets.

Although shallow, lexical, and orthographic features are key components in the goal of improving NE annotation accuracy in the molecular biology domain we believe that future progress towards high accuracy is most likely to come from interaction between machine learning models with constraints (either learnt or given) about the event model of the biological process (for example, cell growth and maintenance, signal transduction, etc.). This should help to resolve many of the contextual ambiguities which often plague NE recognizers in this domain and will be the focus of our future work.

Acknowledgments

This work was supported in part by the Japanese Ministry of Education and Science (Grant No. 14701020) and by a Leadership Fund grant from the National Institute of Informatics. The authors gratefully acknowledge Jun-ichi Tsujii (University of Tokyo) for providing the data set Bio1, Tony Mullen (NII) for his help in setting up experimental scripts, and the anonymous reviewers for their many helpful comments.

References

[1] Sekimizu T, Park H, Tsujii J. Identifying the interaction between genes and gene products based on frequently seen verbs in medline abstracts. In: Genome informatics. Universal Academy Press; 1998. p. 62–71.

[2] Andrade M, Valencia A. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics 1998;14(7):600–7.

[3] Rindflesch TC, Hunter L, Aronson AR. Mining molecular binding terminology from biomedical text. In: American Medical Informatics Association (AMIA)'99 annual symposium, Washington DC, USA; 1999. p. 127–31.

[4] Blaschke C, Andrade C, Ouzounis C, Valencia A. Automatic extraction of biological information from scientific text: protein–protein interactions. Intell Syst Mol Biol 1999;7:60–7.

[5] Collier N, Park H, Ogata N, Tateishi Y, Nobata C, Ohta T, et al. The GENIA project: corpus-based knowledge acquisition and information extraction from genome research papers. In: Proceedings of the annual meeting of the European chapter of the Association for Computational Linguistics (EACL'99), Bergen, Norway; 1999.

[6] Craven M, Kumlien J. Constructing biological knowledge bases by extracting information from text sources. In: Lengauer T, Schneider R, Bork P, Brutlag D, Glasgow J, werner Mewes H, Zimmer R, editors. Proceedings of the seventh international conference on intelligent systemps for molecular biology (ISMB-99), Heidelburg, Germany; 1999. p. 77–86 [ISBN 1-57735-083-9].

[7] Collier N, Nobata C, Tsujii J. Extracting the names of genes and gene products with a hidden Markov model. In: Proceedings of the 18th international conference on computational linguistics (COLING'2000), Saarbrucken, Germany; 2000.

[8] Rindflesch TC, Tanabe L, Weinstein JN, Hunter L. EDGAR: Extraction of drugs, genes and relations from the biomedical literature. In: Pacific symposium on bio-informatics (PSB'2000), Hawai'i, USA; 2000. p. 514–25.

[9] Andrade M, Bork P. Automated extraction of information in molecular biology. FEBS Lett 2000;476(1–2):12–7.

[10] Humphreys K, Demetriou G, Gaizauskas R. Two applications of information extraction to biological science journal articles: enzyme interactions and protein structures. In: Proceedings of the 5th Pacific symposium on biocomputing (BSB 2000), Honolulu, Hawai'i, USA; 2000. p. 505–16.

[11] Collier N, Nobata C, Tsujii J. Automatic acquisition and classification of terminology using a tagged corpus in the molecular biology domain. J Terminol, John Benjamins 2002;7(2):239–57.

[12] Ono T, Hishigaki H, Tanigami A, Takagi T. Automated extraction of information on protein–protein interactions from the biological literature. Bioinformatics 2001;17(2):155–61.

[13] Blaschke C, Hirschman L, Valencia A. Information extraction in molecular biology. Brief Bioinform 2002;3:154–65.

[14] Tanabe L, Wilbur W. Tagging gene and protein names in biomedical text. Bioinformatics 2002;18:1124–32.

[15] Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Rapp BA, Wheeler DL. Genbank. Nucleic Acids Res 2000;28:15–8.

[16] Bairoch A, Apweiler R. The SWISS-PROT protein sequence data bank and its new supplement TrEMBL. Nucleic Acids Res 1997;25:31–6.

[17] Bernstein HM, Koetzle TF, Williams GJ, Meyer EF, Brice MD, Rodgers JR, et al. The protein data bank: a computer-based archival file for macromolecular structures. J Mol Biol 1977;112:535–42.

[18] Berman HM, Battistuz T, Bhat TN, Bluhm WF, Bourne PE, Burkhardt K, et al. The protein data bank. Acta Crystallogr D 2000;58:899–907.

[19] Murzin A, Brenner SE, Hubbard TJP, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–40.

[20] MEDLINE, The PubMed database can be found at: http://www.ncbi.nlm.nih.gov/PubMed/ (1999).

[21] Lindberg DA, Humphreys BL, McCray AT. The unified medical language system. Methods Inform Med 1993;32:281–91.

[22] Lovis C, Michel P, Baud R, Scherrer J. Word segmentation processing: a way to exponentially extend medical dictionaries. Medinfo 1995;8:28–32.

[23] NLM, Medical subject headings. Bethesda, MD: National Library of Medicine; 1997.

[24] Fukuda K, Tsunoda T, Tamura A, Takagi T. Toward information extraction: identifying protein names from biological papers. In: Proceedings of the Pacific symposium on biocomputing'98 (PSB'98); 1998. p. 707–18.

[25] Nobata C, Collier N, Tsujii J. Automatic term identification and classification in biology texts. In: Proceedings of the natural language Pacific rim symposium (NLPRS'2000); 1999.

[26] Kazama J, Makino T, Ohta Y, Tsujii J. Tuning support vector machines for biomedical named entity recognition. In: Workshop on natural language processing in the biomedical domain at the association for computational linguistics (ACL) 2002; 2002. p. 1–8.

[27] Ohta T, Tateishi Y, Mima H, Tsujii J. The GENIA corpus: an annotated research abstract corpus in the molecular biology domain. In: Human language technologies conference (HLT 2002); 2002.

[28] DARPA, Proceedings of the sixth message understanding conference (MUC-6). Columbia, MD, USA: Morgan Kaufmann; 1995.

[29] Tjong Kim Sang EF, De Meulder F. Introduction to the conll-2003 shared task: language-independent named entity recognition. In: Daelemans W, Osborne M, editors. Proceedings of CoNLL-2003, Edmonton, Canada; 2003. p. 142–7.

[30] Vapnik VN. The nature of statistical learning theory. New York: Springer-Verlag; 1995.

[31] Trends and controversies: support vector machines. IEEE Intell Syst 1998:18–28.

[32] Tateishi Y, Ohta T, Collier N, Nobata C, Ibushi K, Tsujii J. Building an annotated corpus in the molecular-biology domain. In: Workshop on semantic annotation and intelligent content, Luxemburg, held in conjunction with COLING'2000; 2000.

[33] Kim JD, Ohta T, Tateishi Y, Tsujii J. GENIA corpus—a semantically annotated corpus for bio-textmining. Bioinformatics 2003;19(Suppl. 1):180–2.

[34] Vapnik VN. The nature of statistical learning theory. New York: Springer-Verlag; 1998.

[35] Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other kernel-based learning methods. Cambridge, England: Cambridge University Press; 2000. [ISBN 0521780195].

[36] Joachims T. Text categorization with support vector machines: learning with many relevant features. In: Proceedings of the European conference on machine learning; 1998.

[37] Kudoh T, Matsumoto Y. Use of support vector learning for chunk identification. In: Proceedings of the fourth conference on natural language learning (CoNLL-2000), Lisbon, Portugal, 2000; p. 142–4.

[38] Nakagawa T, Kudoh T, Matsumoto Y. Unknown word guessing and part-of-speech tagging using support vector machines. In: Proceedings of the sixth natural language processing Pacific rim symposium (NLPRS'2001); 2001. p. 325–31.

[39] Takeuchi K, Collier N. Use of support vector machines in extended named entity recognition. In: Roth D, van den Bosch A, editors. Proceedings of the sixth conference on natural language learning 2002 (CoNLL-2002), Taipei, Taiwan, San Francisco; 2002. p. 119–25.

[40] Isozaki H, Kazawa H. Efficient support vector classifiers for named entity recognition. In: Proceedings of the 19th international conference on computational linguistics (COLING'2002), Taipei, Taiwan; 2002. p. 390–6.

[41] Lee K, Hwang Y, Rim H. Two-phase biomedical NE recognition based on SVMs. In: Proceedings of the ACL workshop on natural language processing in biomedicine, Sapporo, Japan; 2003.

[42] Yamamoto K, Kudo T, Konagaya A, Matsumoto Y. Protein name tagging for biomedical annotation in text. In: Proceedings of the ACL workshop on natural language processing in biomedicine, Sapporo, Japan; 2003. p. 65–72.

[43] Mayfield J, McNamee P, Piatko C. Named entity recognition using hundreds of thousands of features. In: Tjong Kim Sang E, De Meulder FE, editors. Proceedings of the 7th conference on natural language learning 2003 (CoNLL-2003), Edmonton, Canada; 2003.

[44] Noble WS. Support vector machine applications in computational biology. In: Schölkopf B, Tsuda K, Vert J, editors. Kernel methods in computational biology. Cambridge, MA: MIT Press; 2004. [ISBN 0-262-19509-7].

[45] Zien A, Rätsch G, Mika S, Schölkopf B, Lemmen C, Smola A, et al. Engineering support vector machine kernels that recognize translation initiation sites. Bioinformatics 2003;16(9):799–807.

[46] Brown M, Grundy W, Lin D, Cristianini N, Sugnet C, Furey T, et al. Knowledge-based analysis of microarray gene expression data by using support vector machines. Proc Natl Acad Sci USA 2000;97:262–7.

[47] Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learn 2002;46:389–422.

[48] Bock J, Gough D. Predicting protein–protein interactions from primary structure. Bioinformatics 2001;17:455–60.

[49] Hua SJ, Sun ZR. Support vector machine approach for protein subcellular localization prediction. Bioinformatics 2001;17:721–8.

[50] Joachims T. Making large-scale SVM learning practical. In: Scholkopf B, Burges C, Smola A, editors. Advances in kernel methods—support vector learning. Cambridge, MA: MIT Press; 1999. p. 169–84.

[51] Jurafsky D, Martin JH. Speech and language processing—an introduction to natural language processing, computational lingusitics, and speech recognition. New Jersey: Prentice-Hall; 2000. [ISBN 0-13-095069-6].

[52] Brill E. A simple rule-based part of speech tagger. In: Third conference on applied natural language processing (ANLP'92)—association for computational linguistics, Trento, Italy. Cambridge, MA: MIT Press; 1992. p. 152–5.

[53] Tapanainen P, Järvinen T. A non-projective dependency parser. In: Grishman R, editor. Proceedings of the fifth conference on applied natural language processing. Washington Marriot Hotel, Washington, DC: Association of Computational Linguistics; 1997. p. 64–71.

[54] Marcus M, Santorini B, Marcinkiewicz MA. Building a large annotated corpus of English: The Penn Treebank. Comput Linguist 1993;19(2):313–30.

[55] Bikel MB, Schwartz R, Weischendel R. An algorithm that learns what's in a name. Mach Learn 1999;34(1–3):211–31.

[56] Nobata C, Collier N, Tsujii J. Comparison between tagged corpora for the named entity task. In: Kilgarriff A, Sardinha TB, editors. Proceedings of the workshop on comparing corpora at the association for computational linguistics (ACL'2000), Hong Kong; 2000. p. 20–7.

[57] Baluja S, Mittal V, Sukthankar R. Applying machine learning for high performance named-entity extraction. Comput Intell 2000;16(4):586–95.

[58] Zhou G, Zhang J, Su J, Shen D, Tan C. Recognizing names in biomedical texts: a machine learning approach. Bioinformatics 2003;20(7):1178–90.

[59] Chieu HL, Ng HT. Named entity recognition with a maximum entropy approach. In: Proceedings of the seventh conference on natural language learning (CoNLL 2003), Edmonton, Canada; 2003, p. 160–3.

[60] Bikel D, Miller S, Schwartz R, Wesichedel R. Nymble: a high-performance learning name-finder. In: Grishman R, editor. Proceedings of the fifth conference on applied natural language processing (ANLP'97). Washington Marriot Hotel, Washington DC, USA; 1997. p. 194–201.

[61] Hall P, Dowling G. Approximate string matching. Comput Surv 1980;12(4):381–402.

[62] van Rijsbergen CJ. Information retrieval. London: Butterworths; 1979.

[63] Zobel J, Dart P. Phonetic string matching: lessons from information retrieval. In: Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval, Zurich, Switzerland; 1996. p. 166–72.

[64] Nobata C, Collier N, Tsujii J. Comparison between tagged corpora for the named entity task. In: Kilgarriff A, Berber Sardinha, T, editors. Proceedings of the association for computational linguistics (ACL'2000) workshop on comparing corpora, Hong Kong; 2000. p. 20–7.